

2020-09-24 DSpace 7 Working Group Meeting

- [Date](#)
- [Agenda](#)
- [Attendees](#)
- [7.0 Release Goals](#)
- [Current Work](#)
 - [Project Board](#)
 - [Delayed / Needs Discussion](#)
- [Notes](#)

Date

24 Sep 2020 from [14:00-15:00 UTC](#)

Location: <https://lyrasis.zoom.us/my/dspace> (Meeting ID: 502 527 3040).

- More connection options available at [DSpace Meeting Room](#)

Beta 4 Sprint : Ongoing



- Beta 4 active development.
- **Beta 4 PR Deadline of Weds, Sept 23.** Any PRs after that date will not be in Beta 4. Overall goal is to try to release Beta 4 at end of Sept or very early Oct.
- [DSpace 7 Project Boards](#)
- View PRs assigned to you for review/testing: <https://github.com/pulls/review-requested>

Agenda

- **(BEFORE MEETING IN #dev-sprint)** Developer Stand Up - Developers give brief updates on their effort (or their team's effort).
 - Update/see "Current Work" section below based on your status. *Please feel free to update prior to meeting.*
 - Please highlight any new work (needing reviews/testing), any blockers (for you), and any discussion topics you may have.
- (10 mins) General Discussion Topics
 1. *Discussion or Brainstorms on* <https://github.com/DSpace/DSpace/issues/2962#issuecomment-690751137> (This work would occur as part of beta 5)
 - a. Our custom "MultipartFileSender" appears to bypass all of Spring Security (CORS, CSRF, etc) and implements Range headers and ETags in a completely custom manner. This results in CORS issues in Angular for downloads through Angular: <https://github.com/DSpace/DSpace/issues/2962>
 - b. We'd either need to fix this to use Spring tooling, or maybe find a way to proxy downloads (for Scripts & Processes output) to get around CORS.
- (50 mins) Planning for next week
 - Review of our [Beta 4 Project Board](#)

Attendees

- [Art Lowel \(Atmire\)](#)
- [Andrea Bollini \(4Science\)](#)
- [Tim Donohue](#)
- [Lieven Droogmans](#)
- [Giuseppe Digilio \(4Science\)](#)
- [Ben Bosman](#)
- [Paulo Graça](#)
- [Heather Greer Klein](#)

7.0 Release Goals

These resources define the prioritization and general schedule we are working towards

- [DSpace 7 Release Goals](#) : overview of goals/timelines & beta release process
- [DSpace 7 Release Plan spreadsheet](#): our planning spreadsheet which details which features are scheduled for each Beta release.

Current Work

Project Board

DSpace 7.0 Beta 4 Project Board: <https://github.com/orgs/DSpace/projects/3>

To quickly find PRs assigned to you for review, visit <https://github.com/pulls/review-requested> (This is also available in the GitHub header under "Pull Requests Review Requests")

Delayed / Needs Discussion

1. Finalize / approve the initial list of all authorization features which we should implement for the [/api/authz/features REST endpoint](#). This list of features should be limited to *only features which are required to enable/disable User Interface functionality*. (In other words, we can always add more features in the future. We just need to approve the list necessary for 7.0)
 - a. Review current spreadsheet (from [Andrea Bollini \(4Science\)](#)) : https://docs.google.com/spreadsheets/d/1182LcD_WqIZRbUGWpLtBw0aOMR9jhbOVB7GZqtTpR9A/edit?usp=sharing
 - i. [Art Lowel \(Atmire\)](#) : I don't see any immediate issues with the current set of features, but I would prefer a consistent naming scheme. I'd use canDoSomething for everything
 - ii. [Tim Donohue](#) added possible renames of these features based on Art's idea (see cell comments in spreadsheet). I like the "can[DoSomething]" naming scheme as well.
2. Initial Performance Testing from Chris. Needs revisiting / retesting prior to 7.0.
 - a. <https://cwilper.github.io/dspace-perftest/>
 - b. These performance tests were run *prior to* the work on "projections" (to limit the data returned by the REST API). Therefore, it is likely performance is much improved, but needs verification testing.
3. (REST Contract) Edit Homepage News: <https://github.com/DSpace/Rest7Contract/pull/45>
 - a. Delayed. General agreement ([in meeting on March 21, 2019](#)) that storing HTML in metadata fields is not really ideal behavior. Metadata (from a librarian standpoint) tends to be free of format-related markup (as that allows for easier sharing, understanding of metadata. Currently Community & Collection homepage information is HTML-based and *is stored in metadata* that is appropriate for a minor subset of information (like the title) but it is better to move large/rich text to bitstreams.
 - b. Proposal here is to consider storing HTML-based markup (for Site, Community & Collection homepages) in Bitstream(s) associated with the object in question. May allow for more CMS-lite behavior in the future
 - c. Timeline for this is uncertain. Possibly in 7 or 8. May depend on how/whether it can be scoped.

Notes